

Photovoltaic System Design Software

*Professional Software for the Design,
Simulation and Optimization of PV Systems*

Paul DeKleermaeker – N. American General Manager
Valentin Software, Inc.

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Agenda

- Company Overview
- Products Offered
- The PV*SOL Model

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Company Overview

- **Development of Design, Simulation and Modeling Software tools for Photovoltaic and Solar Heating Systems**
- **Established 1988**
- **50 employees (of which over 50% are engineers and developers)**



Global Headquarters:

Dr. Valentin EnergieSoftware GmbH
Berlin, Germany
www.valentin.de

North American Headquarters:

Valentin Software, Inc.
Carlsbad, CA (San Diego area)
www.valentin-software.com
info@valentin-software.com


```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Our Team



```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
        set { m_DesignTemplate = value; }
        get { return m_DesignTemplate; }
    }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Agenda

- Company Overview
- Products Offered
 - Standard Products
 - Company Specific Applications
- The PV*SOL Model

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

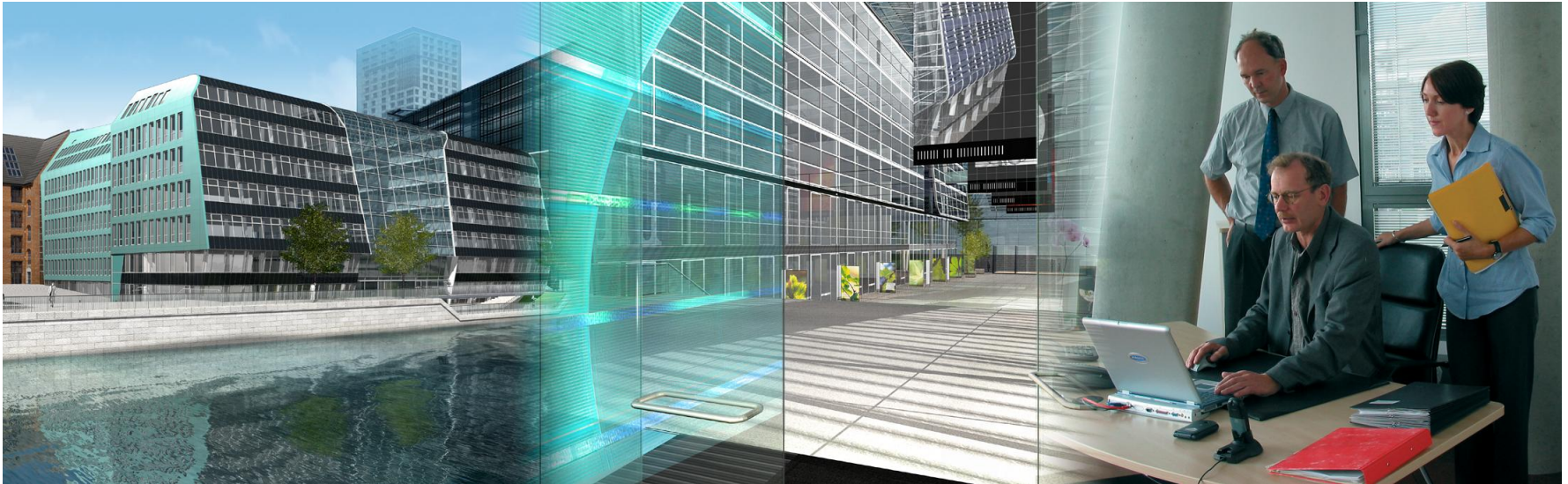
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Business Areas



- Standard Software – *packaged solutions*
- Development of Company Specific Applications– *customized solutions*


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Standard Software Families



PV*SOL Pro

A dynamic simulation program for the professional design and calculation of photovoltaic systems, including grid-connected, off-grid and battery backed-up systems.



T*SOL Pro

A dynamic simulation program for the design and optimization of solar thermal systems, for applications including space heating, domestic hot water, pools and industrial usage.

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
    public string PreviewPrint()
    {
        ReadXML();
        IsPreviewNotRTF = true;
        String sz = GetLLFile();
        Print();
    }

    set { m_DesignTemplate = value; }
    get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Different Levels of Software for Different Levels of Users

- Expert version
- Professional version
- basic version
- Online calculator


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```



PV*SOL basic

PV*SOL basic

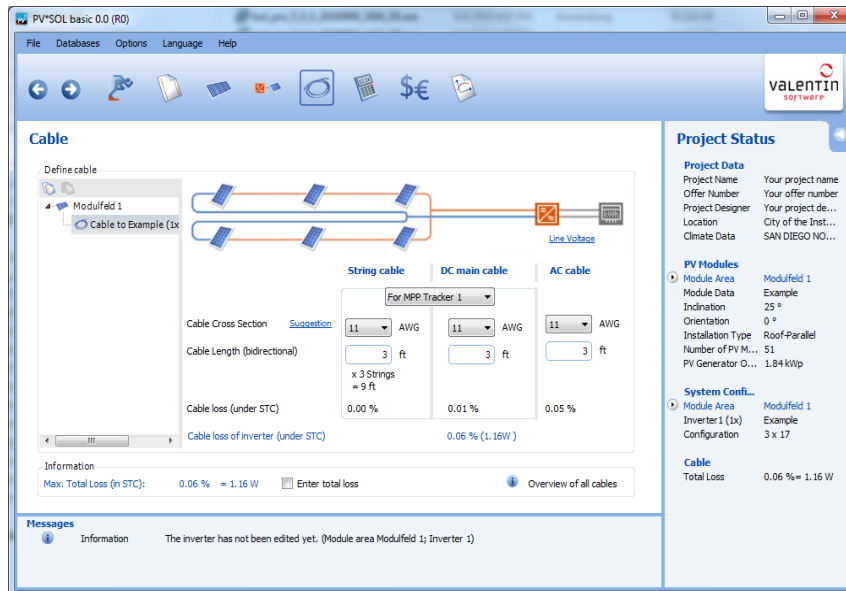
Residential and commercial grid-connected systems, up to 1,000 modules

Automatic inverter selection and configuration

Roof layout

Incentive rates and energy tariffs

Wire size calculation and losses



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

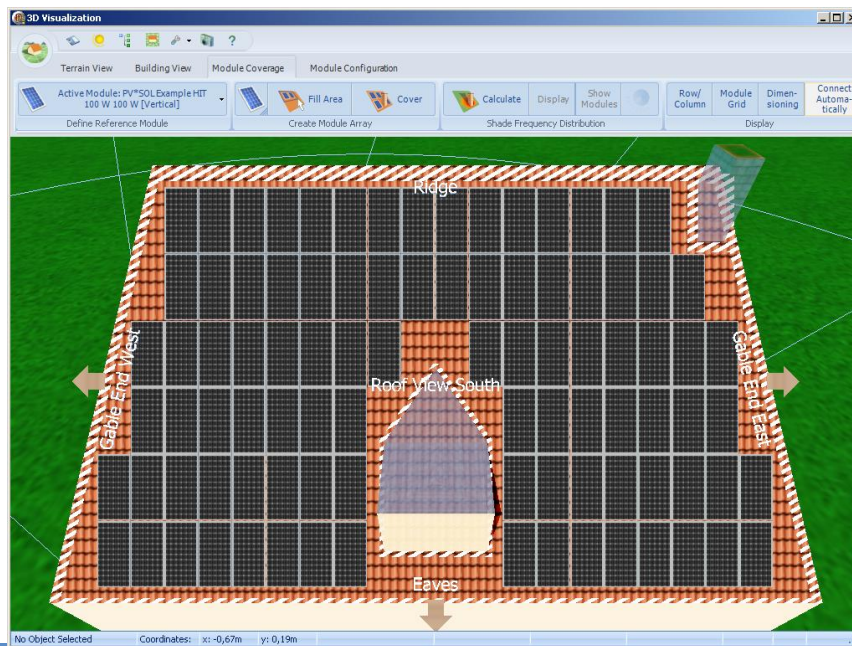


PV*SOL Pro

Residential, commercial and power plant systems up to approx. 100MWp

Grid-connected and off-grid

2D shade analysis



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

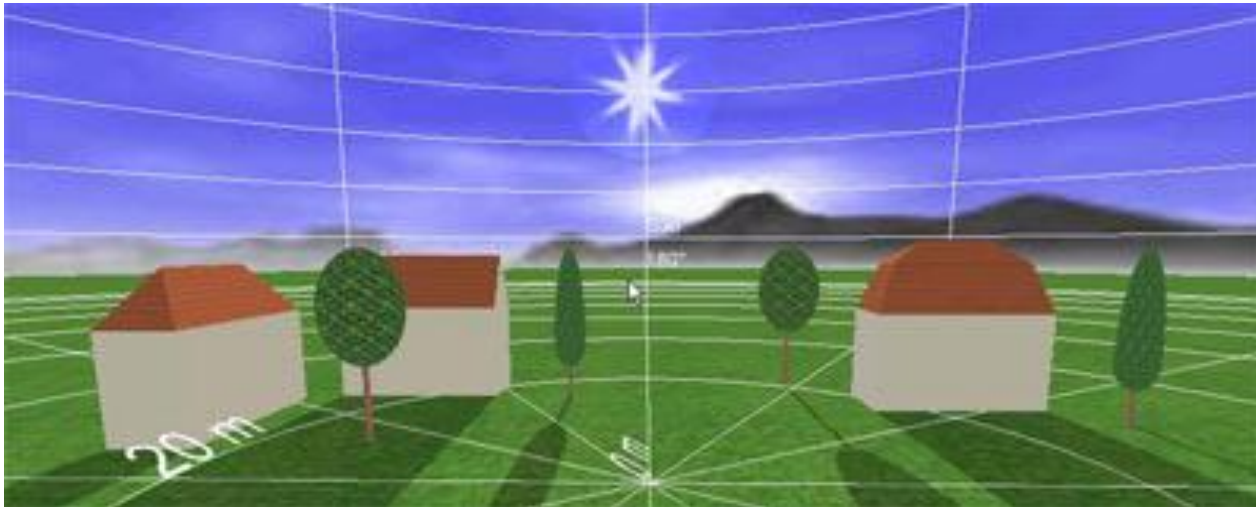
```



PV*SOL Expert

PV*SOL Expert

3D Visualization plus all the capabilities and features of PV*SOL Pro



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

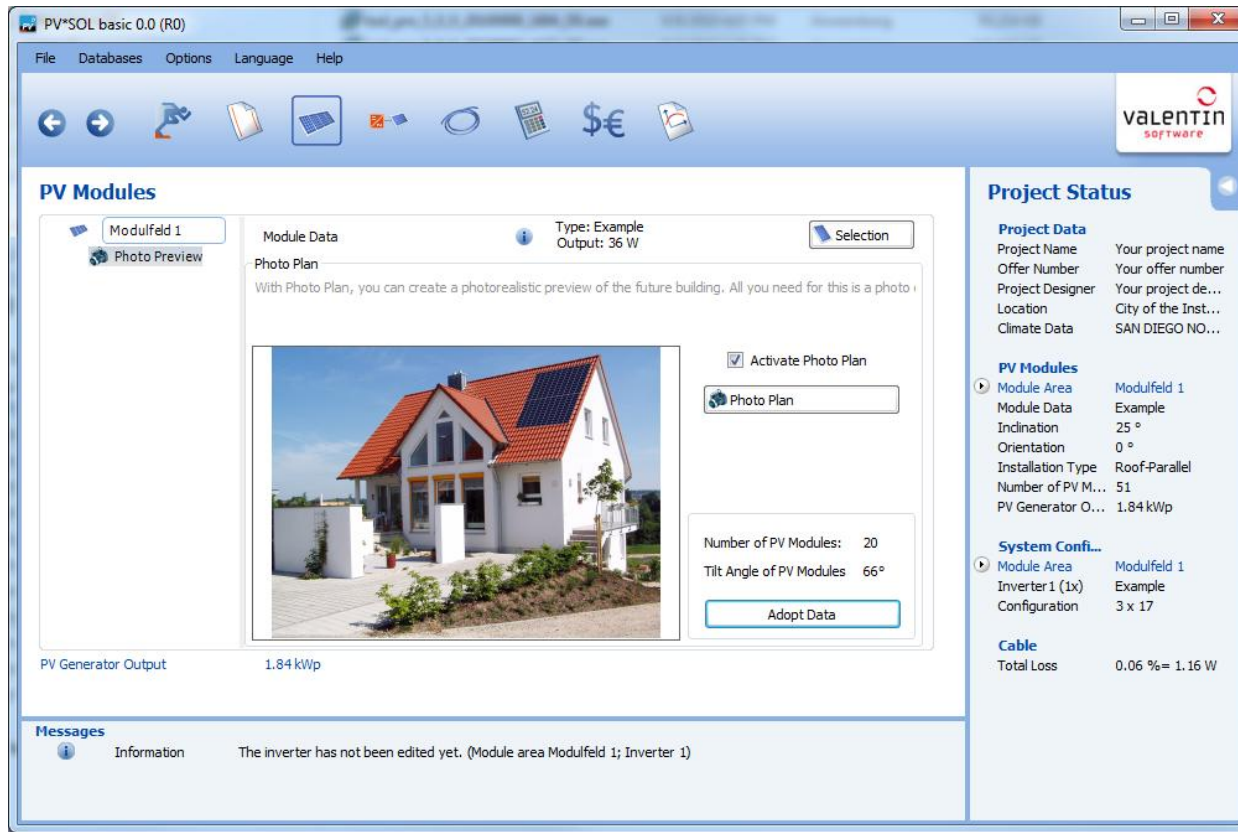
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

PhotoPlan




```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PhotoPlan Output



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

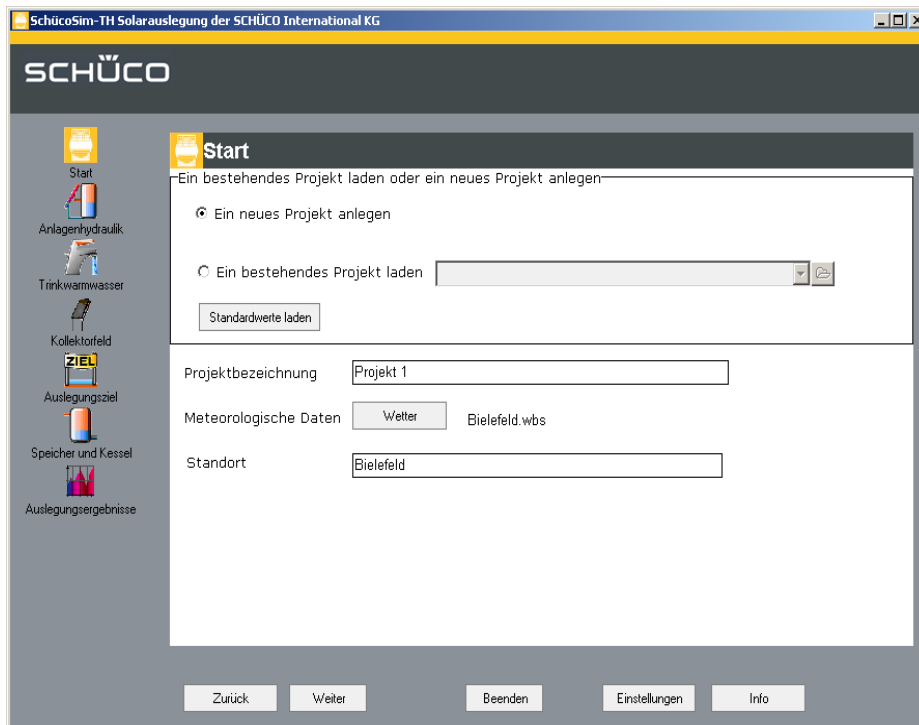
public IntPtr Handle { set; get; }

/// <summary>

```

Company Specific Applications

Customized Programs based on **T*SOL express** or **PV*SOL basic**



- Individual look
- Specific Components
- Design
- Parts List
- End-user Proposals

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
    public string PreviewPrint()
    {
        ReadXML();
        IsPreviewNotRTF = true;
        String sz = GetLLFile();
        Print();
    }

    set { m_DesignTemplate = value; }
    get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Input

- Location for Meteorological Data
- Consumption Profile
- Module Modeling
- Inverter Modeling
- Array Layout
- Losses
- Tariffs

Output

- Energy Balance
- Final Yield
- Efficiency
- Detailed System Data
- Array Layout
- Economic Payback
- Return on Investment

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Developed for wide Market Acceptance

- Robust, proven, stable – customers in over 70 countries, over 70% of installed systems in Germany in 2009 were designed and modeled using PV*SOL.
- Developed for designers, installers, project engineers and developers
- Easy to use: Design assistants, intuitive interface and design flow

Quick Design Grid Connected System Surplus Feed-in

Project Name Solar System Design

Climate Data: San Diego CA

Consumption: Annual Energy Requirement: 4000.0 kWh

PV Array Orientation

Azimuth: 0.00 °

Tilt Angle: 30.00 °

PV System

PV Module: Evergreen Solar ES-180

☒ **Desired System Output** 2.00 kWp ☐ **Set Number of Modules** 11

Inverter: **Inverter Combination**

Sizing

Installed Output [kW] 2.0 **Gross PV Surface Area [m²]** 16.4

Total Number of Modules 11

Inverter	Number of Modules per Inverter	Configuration
1 x SMA Solar Technology AG Sunny Boy SB 2100TL 2.02 kW	11	1 String of 11 Modules

OK Save As Default Technical Data Dialogue >> Simulation Cancel


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

}

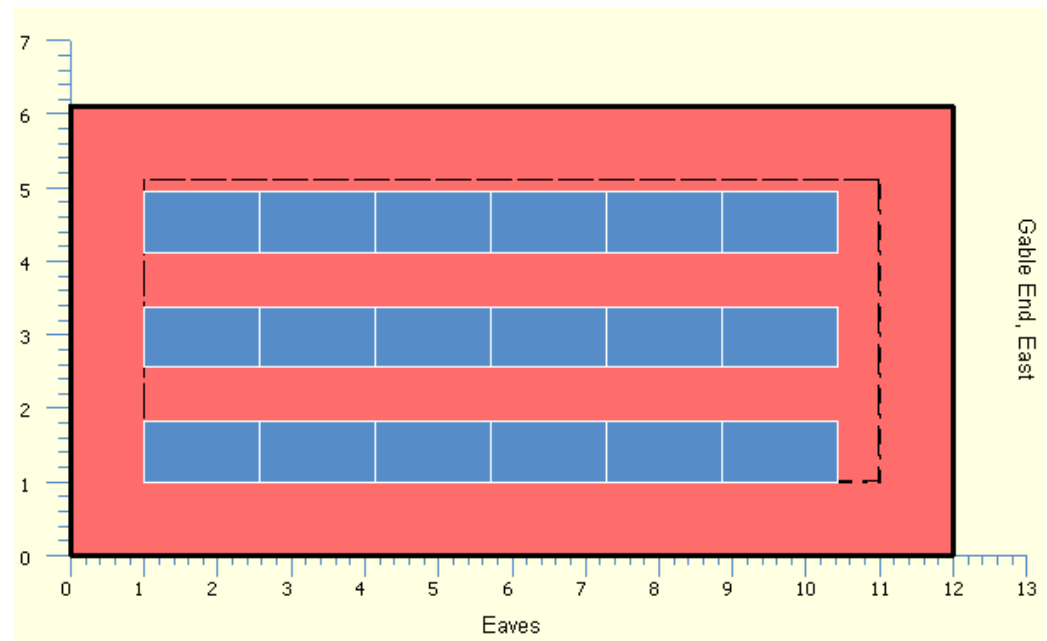
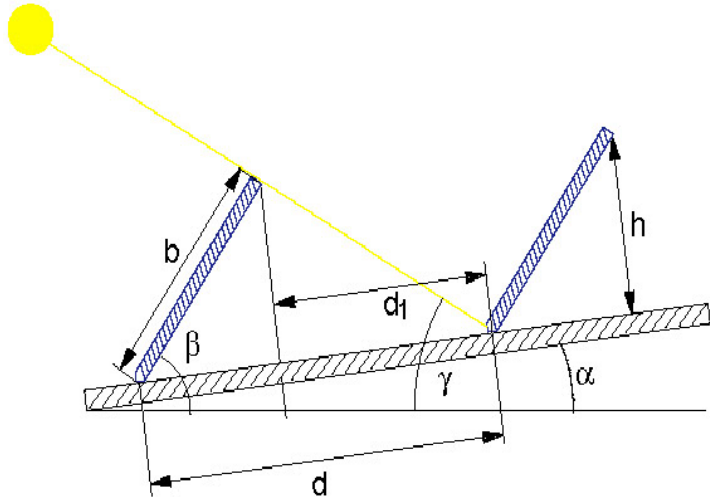
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Roof Parameters and Row Spacing



```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Agenda

- Company Overview
- ProductsOffered
- PV*SOL Model

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

PV*SOL Model

Module Model

- Incident Angle Modifier for reflection
- Module Efficiency Curve for MPP
 - Calculated for all irradiation levels, requires additional partial load operation point
- Complete IV-characteristics for non-MPP operation points
- Efficiency and IV-curves are module temperature corrected
 - 3 temperature coefficients: voltage, current, MPP
- Linear or dynamic temperature model options

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model

Module Technologies

- cSi, aSi, CdTe, CIS, HIT, μ c-Si, Ribbon
- Models each technology's unique characteristics
- Standard data sets for each technology


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

PV*SOL Model

Inverter Model

- Inverter profile and efficiency curve
- Voltage dependent correction
- Ability to model multiple inverters (quantity and type)

Configuration and Automatic Inverter Optimization

- One inverter for multiple sub-arrays with different sizes, modules and orientations
- Multiple types of inverters in one system
- Automatic selection of appropriate configurations
- Automatic inverter sizing and selection

Simulation Frequency

- In hourly increments
- Shade calculated in 10-minute intervals

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String s2 = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

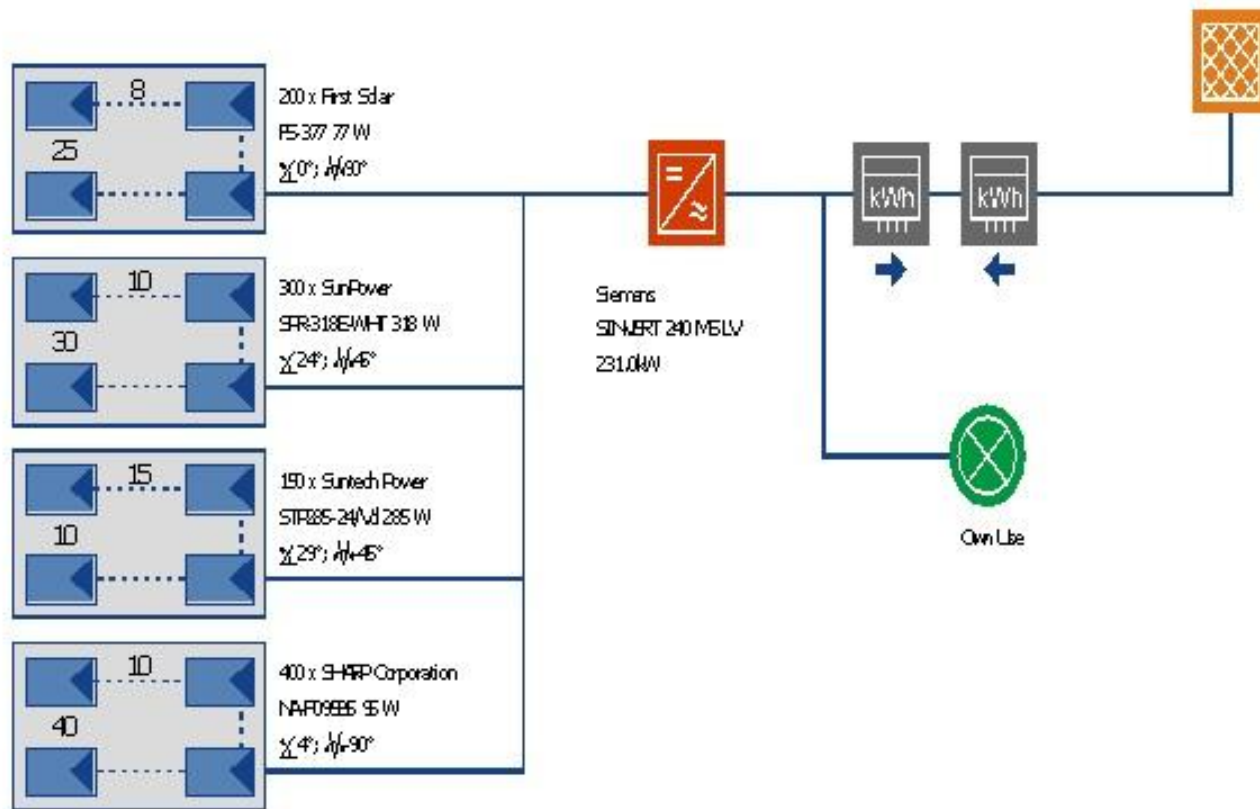
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Diverse Flexibility in Array Design: Mix Technologies, Orientation & Configurations



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Inverter Selection: 1 Inverter, 4 Arrays

pv Load File

Which Inverter should be shown

☒ Matching Default Data ☒ Borderline Cases ☐ not suitable

☐ Filter According to Permissible Unbalanced Load (in relation to total system)

☐ Show Only User-Created Data Records ☐ Show Products that are Not Available

Show All	Manufacturer	Type	No. of MPP Tracks	AC Power Rating [kW]	Unbalanced Load	Max. AC Power [kW]	Min. MPP Voltage [V]	Max. MPP Voltage [V]	Max. Voltage [V]	Max. Current
ACE Ingenieur-Team GbR	Santerno	SUNWAY TG 365DE 600V TE	1	281.4	0.0	309.5	415.0	630.0	740.0	835.6
Advanced Energy Industries	Santerno	SUNWAY TG 300DE 800V TE	1	200.0	0.0	220.0	415.0	760.0	880.0	446.6
AEG Power Solutions GmbH	Santerno	SUNWAY TG 290DE 600V TE	1	221.5	0.0	243.7	415.0	630.0	740.0	657.6
Alcon Elektronik GmbH	Santerno	SUNWAY TG 310DE 800V TE	1	238.6	0.0	262.5	415.0	760.0	880.0	532.8
Alpha Technologies GmbH	Santerno	SUNWAY TG 240DE 800V TE	1	182.7	0.0	201.0	415.0	760.0	880.0	408.0
Ansald Sistemi Industriali	Santerno	SUNWAY TG 175DE 800V TE	1	136.4	0.0	150.0	415.0	760.0	880.0	304.5
Aros	Santerno	SUNWAY TG 385DE 800V TE	1	294.6	0.0	324.0	415.0	760.0	880.0	657.6
ASP AG	Satcon Technology Corpor	PowerGate Plus 250 kW-400V	1	250.0	0.0	250.0	420.0	850.0	900.0	620.0
Bahrman GmbH	Satcon Technology Corpor	PowerGate Plus 250 kW-265V	1	250.0	0.0	250.0	420.0	850.0	900.0	620.0
BONFIGLIOLI VECTRON Gm	Siemens	SINVERT 160 MS LV	1	154.0	0.0	154.0	350.0	750.0	900.0	352.0
Carlo Gavazzi Automation S	Siemens	SINVERT 240 MS LV	1	231.0	0.0	231.0	350.0	750.0	900.0	528.0
CentroSolar AG	Siemens	SINVERT 200 MS - 6DC	1	210.0	0.0	210.0	450.0	750.0	900.0	440.0
Conergy AG	SMA America, Inc.	Sunny Central SC 250U	1	250.0	0.0	250.0	330.0	600.0	600.0	800.0
Danfoss Solar Inverters	SMA Solar Technology AG	Sunny Central SC 250	1	250.0	0.0	250.0	450.0	820.0	880.0	591.0
DCH Solar GmbH	SMA Solar Technology AG	Sunny Central SC 250 HE	1	250.0	0.0	250.0	450.0	820.0	880.0	591.0
Delta Energy Systems	SMA Solar Technology AG	Sunny Central SC 200	1	200.0	0.0	200.0	450.0	820.0	880.0	472.0
Diehl AKO	SMA Solar Technology AG	Sunny Central SC 200 (EVR)	1	200.0	0.0	200.0	450.0	820.0	1000.0	472.0
Dorfmüller	SMA Solar Technology AG	Sunny Central SC 250 (EVR)	1	250.0	0.0	250.0	450.0	820.0	1000.0	591.0
Effekta Regeltechnik GmbH	SMA Solar Technology AG	Sunny Central SC 250 HE (EVR)	1	250.0	0.0	250.0	450.0	820.0	1000.0	591.0
Eltek Valere	Sungrow Power Supply Co	SG250K3	1	250.0	0.0	250.0	450.0	820.0	880.0	600.0
EMERSON Control Technique	Voltwerk electronics GmbH	Voltwerk VC WL 280	1	250.0	0.0	250.0	493.0	780.0	965.0	568.0
Eurotec-electronic Arzberg	Xantrex Technology Inc.	GT250E Grid Tie Inverter	1	250.0	0.0	250.0	450.0	800.0	850.0	555.0
Ever-Solar										
EXENDIS										
FeCon GmbH										
Fortunetree International G										
Fronius International										
Fronius USA										
GH Elektronik GmbH										
GE Consumer & Industrial										
GermanSolar AG										
Grant Engineering (UK) Ltd.										
Green Earth Energy & Tech										
Grosse Elektrotechnik										
HeinTer										

Delete Selection Import Data Record Export Data Record OK Cancel

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Inverter Selection: Multiple and Different Inverters allowed

Configuration Selection

Current PV Module Parameters

Module Type: 90 x Activasun SRS170-24-M 170 W
Total Output: 15,30 [kWp]

Inverter Data

☒ Select Inverter Manufacturer ☐ Select Inverter Type

Manufacturer: SMA America, Inc.

Inverters Included in Selection

Inverter Type 1

Sunny Boy 8000US (USA/277V/60Hz)
Sunny Boy 8000US (USA/240V/60Hz)
Sunny Boy 7000US (USA/208V/60Hz)
Sunny Boy 7000US (USA/240V/60Hz)
Sunny Boy 7000US (USA/277V/60Hz)
Sunny Boy 6000US (USA/208V/60Hz)
Sunny Boy 6000US (USA/240V/60Hz)
Sunny Boy 6000US (USA/277V/60Hz)
Sunny Boy 5000US (USA/208V/60Hz)

Limit Inverter Selection

Number of Different Inverter Types per Configuration: 3

Configuration Criteria

☐ Allow wider Tolerance for Check of Sizing Factors

☒ Show all possible configurations

☐ Filter According to Permissible Unbalanced Load (in relation to total system)

Select Module/Inverter Configuration

No. of Inverters	Number of Inverter Types	Inverter Type 1	Number of Modules per Inverter	Configuration	Unbalanced Load	Sizing Factor[%]
3	1	3 x SMA America, Inc. Sunny Boy 5000US (USA/208V/60Hz) 5 kW	30	3 Strings x 10 Modules	0,00	102,0
3	1	3 x SMA America, Inc. Sunny Boy 6000US (USA/240V/60Hz) 6 kW	30	3 Strings x 10 Modules	0,00	85,0
3	1	3 x SMA America, Inc. Sunny Boy 6000US (USA/277V/60Hz) 6 kW	30	3 Strings x 10 Modules	0,00	85,0
3	1	3 x SMA America, Inc. Sunny Boy 6000US (USA/208V/60Hz) 6 kW	30	3 Strings x 10 Modules	0,00	85,0
3	2	1 x SMA America, Inc. Sunny Boy 7000US (USA/277V/60Hz) 7 kW 2 x SMA America, Inc. Sunny Boy 3000US (USA/240V/60Hz) 3 kW	48 21	4 Strings x 12 Modules 3 Strings x 7 Modules	4,00	116,6 119,0
3	2	1 x SMA America, Inc. Sunny Boy 7000US (USA/277V/60Hz) 7 kW 2 x SMA America, Inc. Sunny Boy 3000US (USA/208V/60Hz) 3 kW	48 21	4 Strings x 12 Modules 3 Strings x 7 Modules	4,00	116,6 119,0
3	2	1 x SMA America, Inc. Sunny Boy 7000US (USA/240V/60Hz) 7 kW 2 x SMA America, Inc. Sunny Boy 3000US (USA/208V/60Hz) 3 kW	48 21	4 Strings x 12 Modules 3 Strings x 7 Modules	4,00	116,6 119,0
3	2	1 x SMA America, Inc. Sunny Boy 7000US (USA/208V/60Hz) 7 kW 2 x SMA America, Inc. Sunny Boy 3000US (USA/277V/60Hz) 3 kW	48 21	4 Strings x 12 Modules 3 Strings x 7 Modules	4,00	116,6 119,0

Max. Number of Configurations Shown: 50 (Number of Calculated Configurations: 7257)

OK Cancel


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

System Check before Simulation

System Check

Output Check		Upper Voltage Threshold Check	
PV Output per Inverter:	192 kW	Inverter Max. System Voltage:	900 V
Inverter AC Power Rating:	231 kW	Module Max. System Voltage:	
Sizing Factor: (PV Output (STC) AC Power Rating)	83 %	In Array 1:	1000 V
		In Array 2:	1000 V
		In Array 3:	1000 V
Permissible Sizing Factor:	80 % - 120 %	In Array 4:	1000 V
MPP Voltage Check		PV Array Open Circuit Voltage:	
Inverter MPP Tracking Range:	350 - 750 V	In 1. Array (A1 = -10°C u. 1000 W/m²):	539 V
PV Array MPP Voltage at (A1 + 1000 W/m²) or (15 °C + G2):		In 2. Array (A1 = -10°C u. 1000 W/m²):	709 V
In 1. Array (A1 = 70°C; G2= 1000 W/m²):	351 - 415 V	In 3. Array (A1 = -10°C u. 1000 W/m²):	752 V
In 2. Array (A1 = 70°C; G2= 1000 W/m²):	461 - 566 V	In 4. Array (A1 = -10°C u. 1000 W/m²):	721 V
In 3. Array (A1 = 70°C; G2= 1000 W/m²):	434 - 560 V		
In 4. Array (A1 = 70°C; G2= 1000 W/m²):	411 - 490 V		

Unbalanced Load Check

Current Unbalanced Load:	0.0 kVA	Maximum Permissible Unbalanced Load:	4.6 kVA
--------------------------	---------	--------------------------------------	---------

No discrepancies found!
Please observe any design recommendations made by the manufacturer.

Total System Cabling Lossess

Calculations with set extremes (see Options->Settings) Continue Help

System Check

Array 1		Array 2	
Current through Cabling under STC:	39 A	Current through Cabling under STC:	175 A
Max. Capacity of Insulated Copper Wiring, Group C:	1125 A	Max. Capacity of Insulated Copper Wiring, Group C:	1350 A
Rel. Cabling Losses under STC:	0.033 %	Rel. Cabling Losses under STC:	0.093 %
max. Current through Inverter at 25 °C and 1000 W/m²	373 A	max. Current through Inverter at 25 °C and 1000 W/m²	373 A
Max. Inverter Input Current:	528 A	Max. Inverter Input Current:	528 A
Array 3		Array 4	
Current through Cabling under STC:	79 A	Current through Cabling under STC:	80 A
Max. Capacity of Insulated Copper Wiring, Group C:	450 A	Max. Capacity of Insulated Copper Wiring, Group C:	1800 A
Rel. Cabling Losses under STC:	0.130 %	Rel. Cabling Losses under STC:	0.037 %
max. Current through Inverter at 25 °C and 1000 W/m²	373 A	max. Current through Inverter at 25 °C and 1000 W/m²	373 A
Max. Inverter Input Current:	528 A	Max. Inverter Input Current:	528 A

Unbalanced Load Check

Current Unbalanced Load:	0.0 kVA	Maximum Permissible Unbalanced Load:	4.6 kVA
--------------------------	---------	--------------------------------------	---------

No discrepancies found!
Please observe any design recommendations made by the manufacturer.

Total System Cabling Lossess

Calculations with set extremes (see Options->Settings) Continue Help

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model - Climate

Weather Data Source

- MeteoSyn, Meteonorm, TMY3, SWERA, PVGIS, NASA-SSE, user import and create

Irradiance Model

- Hay and Davies model
- Albedo - monthly

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model - Shading

Shading

- Horizon based
 - user-defined or imported
- 3D Model
 - Area based
 - Models near and horizon shade in 10 minute intervals
 - Calculates impact per string of cells within each module

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
```

```
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Expert 3D Visualization




```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

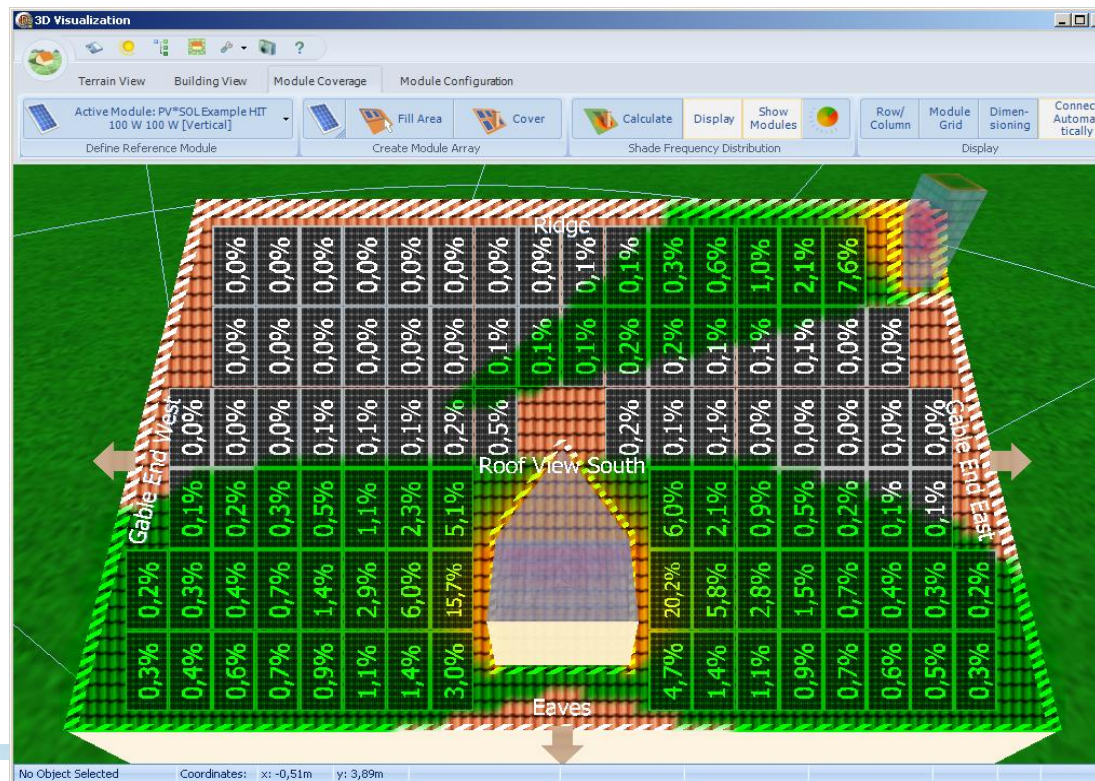
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Shade Analysis and Degradation



```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

String Optimization



```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model - Losses

Losses & Derate Parameters

- Module mismatch, diodes & module quality
- Wiring – calculated from cable data
- Deviation from standard spectrum
- Soiling


```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model – Financial Analysis

System Financials

- Economic efficiency and cash-flow analysis
- User defined complex tariffs and incentives
- Loans


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

User Defined Tariff Rates

Tariff Periods Definition

Winter Tariff Period - From: To:

☐ Tariff Periods Summer/Winter Different

☒ Define Special Tariff (ST) Periods

On which day/s are new Tariff Periods valid?

☐ Tue ☐ Wed ☐ Thu ☐ Fri ☒ Sat ☒ Sun

HT Periods

	From	To	From	To	From	To
Mon - Fri	6:00	12:00	14:00	18:00	0:00	0:00
Sat	0:00	0:00	0:00	0:00	0:00	0:00
Sun	0:00	0:00	0:00	0:00	0:00	0:00

HT Periods / ST Periods

OK Cancel

Kilowatt-Hour Price

☐ Summer/Winter Different

☒ Price Bands for Each Tariff Zone Different

Price Change Factor

Data for Total HT Consumption

	[kWh]	[cents/kWh]
From	0	14.00
Each further kWh from	1,000	18.00
Each further kWh from	2,000	24.00
Each further kWh from	3,000	48.00
Each further kWh from		
Each further kWh from		
Each further kWh from		

kWh Price - HT / kWh Price - LT

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>

```

```

public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

```

```

    set { m_DesignTemplate = value; }
    get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

PV*SOL basic 0.0 (R0)

File Databases Options Language Help

Financial Analysis

Type of project ☒ Residential ☐ Commercial

System costs

Total installed system cost € ☒ Payed

Yearly operation and maintainance cost €/year Expected system life years

Inverter replacement cost € Due every years

Consumption

Utility Utility rate

☒ Annual electricity bill € ☐ Monthly load

Yearly increase of energy costs % [Estimate with Home Energy Saver](#)

Information relevant for incentives selection

Yearly income € Tax status

☐ Choose second option of incentives where applicable.

Project Status

Project Data

Project Name A new residenti...
 Proposal Number Your proposal n...
 Project Designer Your project de...
 Project Location Ihr Projektstan...
 Climate Data San Diego Ca

PV Modules

Module Area Your Module Area

Module Data Example
 Tilt 25 °
 Orientation 0 °
 Mounting Type Roof-Parallel
 Number of PV M... 51
 PV System DC ... 1,84 kWp

System Confi...

Module Area Your Module Area

Inverter1 (1x) Example
 Configuration 3 x 17

Cable

Total Loss 0,37%= 6,79 W

Messages

Information The inverter has not been edited yet. (module area Your Module Area; inverter 1)

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

```

```

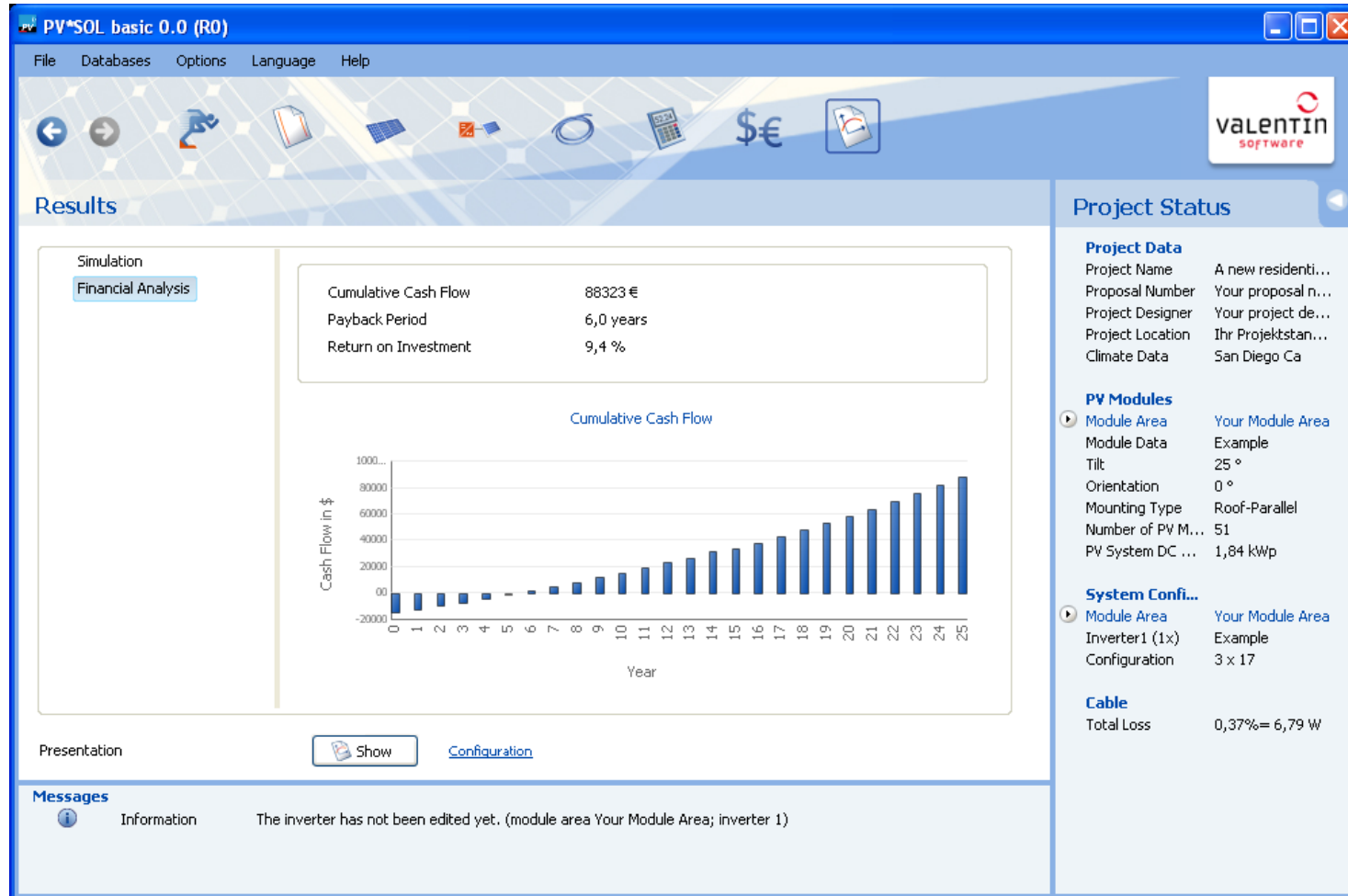
    set { m_DesignTemplate = value; }
    get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```



```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}
```

```
set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model – Project Results

Report

- Location, climate data, equipment, load, array, inverter, production, losses
- Array layout
- 3D Views or photo realistic
- Financial Analysis

Output options

- Graphical
- Table format
- Export, copy/paste
- .pdf


```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Output Options

Select Curve

Radiation [W/m²]

- ☒ Specific Irradiation onto Horizontal
- ☒ Specific PV Array Irradiation
- ☒ Specific PV Array Irradiation minus Reflection

Wind [m/s]

- ☒ Wind Speed at Height of 10m

Temperature [°C]

- ☒ External Temperature

Select Curve

Climate Data

Selected Data

- ☒ PV Array Irradiation
- ☐ Energy Produced by PV Array (AC)
- ☐ Energy to Grid
- ☐ Direct Use of PV Energy
- ☐ Energy from Grid
- ☐ Energy Produced by PV Array
- ☐ PV System's own Energy Consumption

Climate Data **Energy** Total System

Selected Data Records: 8

Des

Select Curve

Evaluation Measurements

- ☒ Solar Fraction
- ☐ System Efficiency

Efficiency

- ☐ Inverter Efficiency
- ☐ PV Array Efficiency

☒ Module Temperature

Climate Data **Energy** Total System

Selected Data Records: 8

Deselect OK Cancel

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

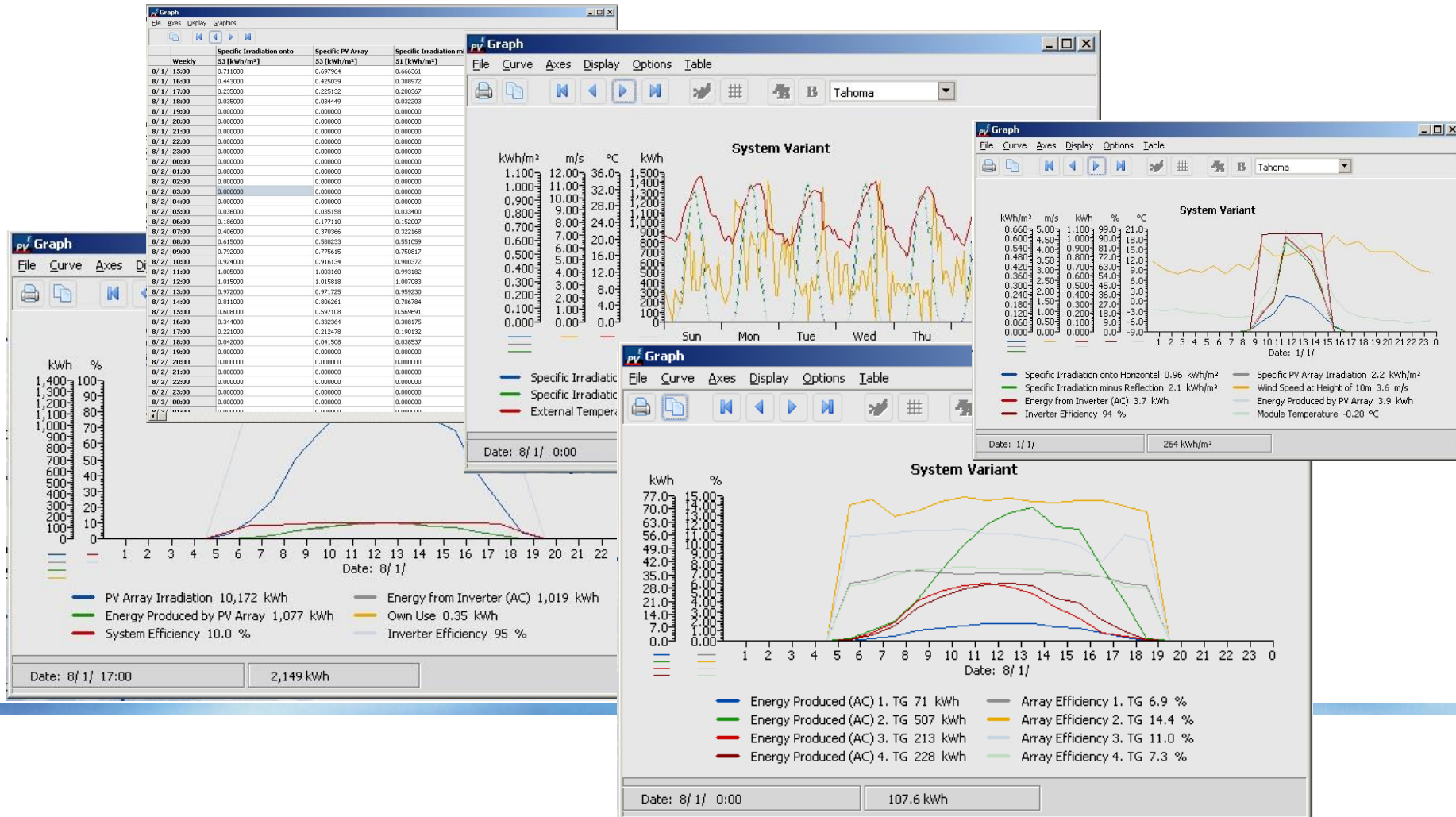
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Detailed Reporting Options



```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

PV*SOL Model - Databases

Component

- Over 5,000 modules
- Over 1,200 inverters
- User can create or modify
- Automatic updates via internet, distributed ~ weekly

Climate

- 1,020 US TMY3 locations
- 8,000 global locations
- User can create and import

```

/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

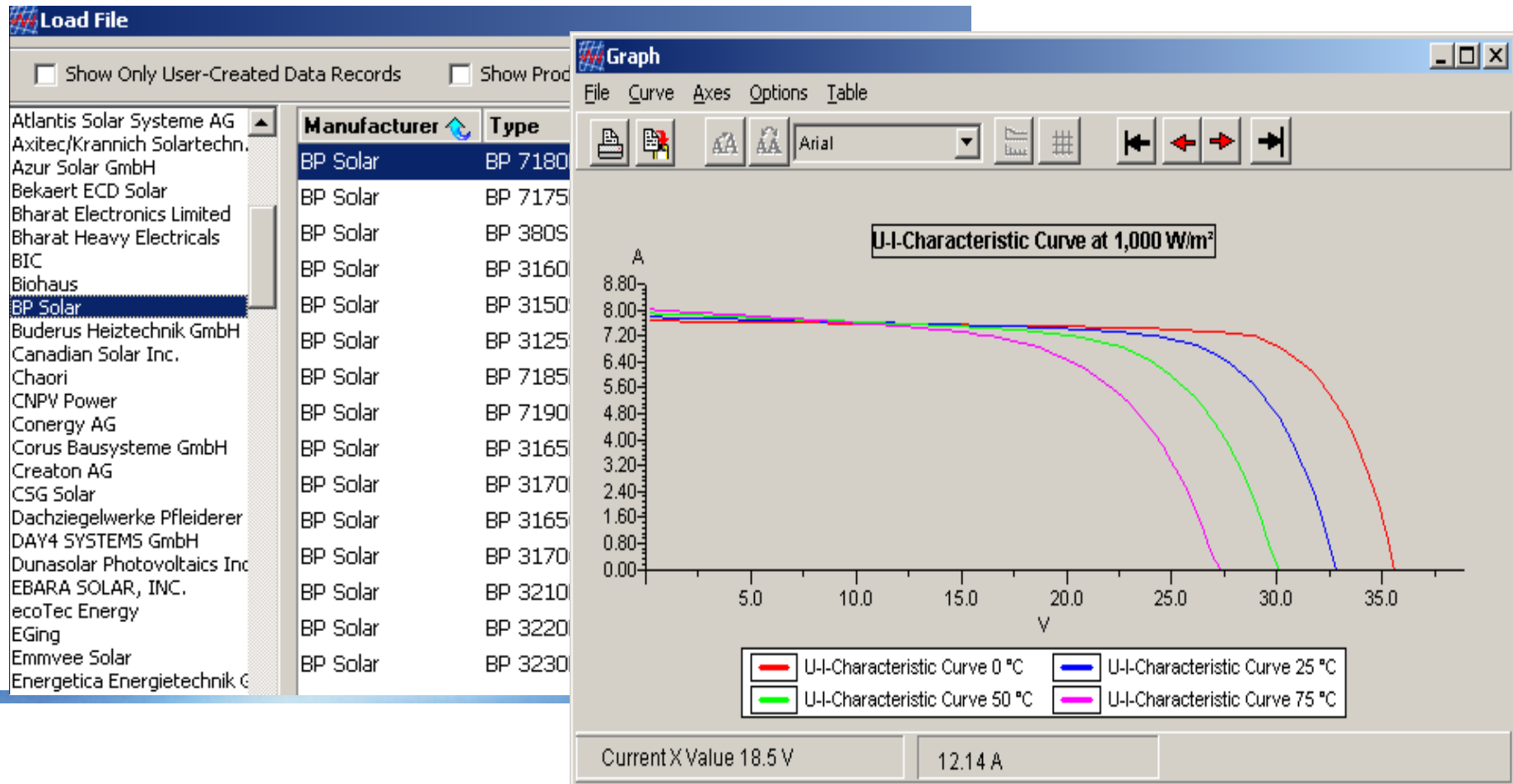
/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>

```

Module Database




```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
public string PreviewPrint()
{
    ReadXML();
    IsPreviewNotRTF = true;
    String sz = GetLLFile();
    Print();
}

set { m_DesignTemplate = value; }
get { return m_DesignTemplate; }

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```



Download Free Demo Software

Free demo software of both PV*SOL and
T*SOL are available at:

www.valentin-software.com

```
/// <summary>
/// Fill the variables
/// </summary>
/// <returns></returns>
    public string PreviewPrint()
    {
        ReadXML();
        IsPreviewNotRTF = true;
        String sz = GetLLFile();
        Print();
    }

    set { m_DesignTemplate = value; }
    get { return m_DesignTemplate; }
}

/// <summary>
/// Sets the language for the Preview Editor (Toolbar)
/// </summary>
public String Language { set; get; }

public IntPtr Handle { set; get; }

/// <summary>
```

Thank You for Your Attention



Standalone System in Nepal
Designed by PV*SOL